Helm 3 Deep Dive
All the things you never wanted to know
A Huge Thanks

- The Helm Core maintainers
- Josh Dolitsky
BRIEF HISTORY OF HELM

- "k8splace" is developed at a Deis hackathon (OCT 2015)
- Helm v1 is announced at the first KubeCon (NOV 2015)
- Helm merges with Google Deployment Manager (JAN 2016)
- Helm 2.0.0 is released (NOV 2016)
- Helm Summit takes place to kick off discussion on Helm 3 (FEB 2018)
- Helm project joins the CNCF (JUN 2018)
- Helm 3 is Released (???)

SOON.
The Helm 3 Project Board

https://github.com/helm/helm/projects/1
Major changes to expect in Helm 3

- Embedded Lua engine
- Removal of Tiller
- Chart repo auth & upload
- New Charts.yaml
Major changes to expect in Helm 3

- Embedded Lua engine
- Removal of Tiller
- Chart repo auth & upload
- New Charts.yaml
Major changes to expect in Helm 3

- Embedded Lua engine
- Removal of Tiller
- Chart repo auth & upload
- New Charts.yaml
Major changes to expect in Helm 3

- Embedded Lua engine
- Removal of Tiller
- Chart repo auth & upload

- New Chart.yaml
Other Helm 3 changes

- “Managed” hooks - if Helm creates something, it will delete it
- index.yaml will move to index.json, and be partitioned for performance
- Schematize your values by including a values.schema.yaml file
- Helm client libraries will be much better
Embedded Lua engine

Making Helm charts more robust and programmable
Major changes to expect in Helm 3

**Embedded Lua engine**

- Lua can be used with, or in place of, YAML templates
- Treat Kubernetes resources as objects, not strings
- Intercept lifecycle events, modify chart on-the-fly
- Sandboxed - optionally limit access to network/io
- Develop and import reusable “library charts”
- Helm plugins will have access to the Lua runtime

Removal of Tiller

Chart repo auth & upload

New Chart.yaml
apiVersion: v1
kind: Pod
metadata:
  name: {{ template "alpine.fullname" . }}
labels:
  heritage: {{ .Release.Service }}
  release: {{ .Release.Name }}
  app: {{ template "alpine.name" . }}
spec:
  restartPolicy: {{ .Values.restartPolicy }}
  containers:
  - name: waiter
    image: "{{.Values.img}}:{{.Values.img.tag}}"
    imagePullPolicy: {{ .Values.img.pullPolicy }}
    command: ["/bin/sleep", "9000"]
apiVersion: v1
kind: Pod
metadata:
  name: template "alpine.fullname" .
labels:
  heritage: template .Release.Service 
  release: .Release.Name 
  app: template "alpine.name" .
spec:
  restartPolicy: .Values.restartPolicy 
  containers:
    - name: waiter
      image: .Values.img .. .Values.img.tag
      imagePullPolicy: .Values.img.pullPolicy 
      command: [/bin/sleep", "9000"]

---

function create_alpine_pod(_)
local pod = {
  apiVersion = "v1",
  kind = "Pod",
  metadata = {
    name = alpine_fullname(_),
    labels = {
      heritage = _.Release.Service or "helm",
      release = _.Release.Name,
      chart = _.Chart.Name .. "-" .. _.Chart.Version,
      app = alpine_name(_) 
    }
  },
  spec = {
    restartPolicy = _.Values.restartPolicy,
    containers = {
      name = waiter,
      image = _.Values.image.repository .. ":",
      imagePullPolicy = _.Values.image.pullPolicy 
      command = {
        "/bin/sleep", "9000"
Helm 2

apiVersion: v1
dKind: Pod
metadata:
  name: {{ template "alpine.fullname" . }}
  labels:
    heritage: {{ .Release.Service }}
    release: {{ .Release.Name }}
    app: {{ template "alpine.name" . }}
spec:
  restartPolicy: {{ .Values.restartPolicy }}
  containers:
    - name: waiter
      image: "{{ .Values.img }}:{{ .Values.img.tag }}"
      imagePullPolicy: {{ .Values.img.pullPolicy }}
      command: ["/bin/sleep", "9000"]

Helm 3

-- Example of using a "library chart"
local pods = require("mylib.pods");

function create_alpine_pod(_)
    myPod = pods.new("alpine:3.7", _)
    myPod.spec.restartPolicy = "Always"
    -- set any other properties
    _.Manifests.add(myPod)
end
Removal of Tiller

Improving security by delegating auth to Kubernetes RBAC
Removal of Tiller

- Shrinks the security model for Helm, now client-only
- Auth is delegated to Kubernetes RBAC
- Release history maintained using ULIDs vs. integers
- “Release” CRD will store instance of an application
- “ReleaseVersion” CRD will store version of release
Chart repo auth & upload

New commands and API spec for working with chart repositories
Major changes to expect in Helm 3

**Chart repo auth & upload**

- “helm push” command to upload chart to a repo
- Push spec is still being finalized
- Plugins can supply custom protocols (e.g. `s3://`)
- “helm login” command to authenticate against a repo
- OAuth2 authorization flow, bearer/token auth
- Limit which users can upload/install which charts

**Embedded Lua engine**

**Removal of Tiller**

**Chart repo auth & upload**

**New Chart.yaml**
```bash
$ helm repo add myrepo https://site.com/myrepo --username=josh --password=*****
$ helm package mychart/
$ ./my-custom-uploader.sh mychart-0.1.0.tgz
$ helm repo update
$ helm install myrepo/mychart
```
$helm repo add myrepo  
  https://site.com/myrepo  
  --username=josh  
  --password=*****

$helm login https://site.com

$helm repo add myrepo  
  https://site.com/myrepo

$helm push mychart/ myrepo

$helm repo update

$helm repo add myrepo  
  https://site.com/myrepo

$helm push mychart/ myrepo

$helm repo update

$helm install myrepo/mychart
New chart.yaml

- Current chart.yaml files will not be broken
- The requirements file will no longer exist
- Requirements will now be part of the charts.yaml
# requirements.yaml

dependencies:
- name: requsubchart
  version: 0.1.0
  repository: "https://example.com/charts"
- name: requsubchart2
  version: 0.2.0
  repository: "https://example.com/charts"
- name: requsubchart3
  version: ">=0.1.0"
  repository: "https://example.com/charts"

# chart.yaml

description: A Helm chart for Kubernetes
name: requtest
version: 0.1.0
Helm 2

# requirements.yaml
dependencies:
- name: reqsubchart
  version: 0.1.0
  repository: "https://example.com/charts"
- name: reqsubchart2
  version: 0.2.0
  repository: "https://example.com/charts"
- name: reqsubchart3
  version: ">=0.1.0"
  repository: "https://example.com/charts"

# chart.yaml

description: A Helm chart for Kubernetes
name: reqtest
version: 0.1.0

Helm 3

description: A Helm chart for Kubernetes
name: reqtest
version: 0.1.0

dependencies:
- name: reqsubchart
  version: 0.1.0
  repository: "https://example.com/charts"
- name: reqsubchart2
  version: 0.2.0
  repository: "https://example.com/charts"
- name: reqsubchart3
  version: ">=0.1.0"
  repository: "https://example.com/charts"
Want to know more about Helm 3?

**Helm 3 Design Proposal**
[https://github.com/helm/community/tree/master/helm-v3](https://github.com/helm/community/tree/master/helm-v3)

**The Helm 3 Dev Branch**
[https://github.com/helm/helm/tree/dev-v3](https://github.com/helm/helm/tree/dev-v3)